Satisfiability Planning with Causal Theories

Norman McCain and **Hudson Turner**
Department of Computer Sciences
University of Texas at Austin
{mccain,hudson}@cs.utexas.edu

Abstract

A previous paper introduced the nonmonotonic formalism of *causal theories*, along with a general method for representing action domains in it. Here we show that causal action theories provide a basis for effective automated planning. To this end, we define several properties plans may have, such as executability, determinism, and validity. We then identify a class of causal action theories for which (i) there is a concise translation into classical logic, and (ii) the models of the resulting classical theories correspond to valid plans. These results enable satisfiability planning (in the sense of Kautz and Selman) on the basis of action formalizations that include indirect effects of actions (ramifications), implied action preconditions (qualifications), concurrent actions, and other features of action domains. Causal theories representing the large blocks world and logistics planning problems from [Kautz and Selman, 1996] are solved comparatively quickly, demonstrating the effectiveness of our approach.

1 INTRODUCTION

The language of causal theories is a simple nonmonotonic formalism, intended for representing the conditions under which facts are caused. The formalism was introduced in [McCain and Turner, 1997], along with a general method for representing action domains as causal theories. In the current paper, we describe an implemented approach to satisfiability planning [Kautz and Selman, 1992, 1996], which is based on a translation from the "definite" subclass of causal theories into classical propositional logic. This approach to planning is noteworthy for two reasons. First, it is based on a formalism for describing action domains that is more expressive than the STRIPS-based formalisms traditionally used in automated planning. Such features, for example, as indirect effects of actions (ramifications), implied action preconditions (qualifications), and concurrent actions are easily represented in causal theories.[1] Secondly, our experiments suggest that the additional expressiveness of causal theories comes with no performance penalty in satisfiability planning. Specifically, in this paper we show that the large blocks world and logistics planning problems used by Kautz and Selman [1996] to demonstrate the effectiveness of satisfiability planning can be conveniently represented as causal theories and solved in times comparable to those that they have obtained.

Because causal theories are more expressive than traditional planning languages, we must consider the preliminary question of when a sequence of actions is a valid plan for achieving a goal G in an initial situation S_0. A valid plan has two fundamental properties: sufficiency and executability. Roughly speaking, a sufficient plan will always achieve G if carried out starting in S_0, and an executable plan can always be carried out starting in S_0. We will make these ideas precise, in the setting of causal action theories.

We must also consider how to find valid plans by the satisfiability method. Assume that T is a classical propositional theory describing the worlds that are "causally possible" for an action domain. In satisfiability planning, a plan is obtained by extracting the sequence of actions from a model of T that satisfies both the initial state S_0 and the goal G. We will call a plan obtained in this way a causally possible plan, because what we know in this case is simply that there is at least one causally possible world in which the

[1] For more on the expressiveness of causal theories, see [McCain and Turner, 1997, Giunchiglia and Lifschitz, 1998, Lifschitz, 1997, McCain, 1997, Turner, 1998].

plan achieves G starting in S_0. In order for satisfiability planning to be sound, we must guarantee that the causally possible plans are in fact valid. Accordingly, we define a subclass of definite causal theories, called "simple," and show that their translations into classical logic are suitable for satisfiability planning. That is, the plans obtained from the models of their translations are not only causally possible, but also deterministic, and thus, as we will show, valid.

The main contributions of the paper are (1) to provide a theoretical foundation for satisfiability planning on the basis of causal theories, and (2) to present experimental evidence that the approach is relatively effective. More specifically, we define a family of fundamental properties a plan may have: causally possible, deterministic, sufficient, executable. We say a plan is valid if and only if it is sufficient and executable. We prove that every causally possible, deterministic plan is valid. We then identify a class of "simple" causal theories suitable for satisfiability planning. Simple causal theories have a concise translation into classical logic, and, as we prove, the classical models yield valid plans. Simple causal theories are very expressive, thus enabling planning with respect to a wide variety of action domains. We also provide experimental evidence that this planning approach can be very effective on classical problems, by solving, comparatively quickly, the large blocks worlds and logistics planning problems from [Kautz and Selman, 1996].

The paper is organized as follows. Section 2 reviews the syntax and semantics of causal theories, and defines a concise translation from definite causal theories into classical logic. Section 3 illustrates our method of formalizing action domains in causal theories. Section 4 defines plan validity and related notions for causal action theories. Section 5 defines the class of simple causal theories, and presents the main theorem showing that satisfiability planning is sound for simple theories. Section 6 describes an implementation of satisfiability planning with causal theories. Section 7 reports experimental results on the large blocks world and logistics planning problems from [Kautz and Selman, 1996]. Section 8 consists of the proof of the main theorem. We conclude briefly in Section 9.

2 CAUSAL THEORIES

2.1 SYNTAX

Begin with a language of propositional logic, whose signature is given by a nonempty set of atoms. (In application to formalizing action domains, atoms are taken to represent propositions about the values of fluents and the occurrences of actions at specific times.) Assume that the language includes a zero-place logical connective *True* such that *True* is a tautology. Let *False* stand for $\neg$ *True*. A *literal* is an atom or the negation of an atom. For any literal L, $\overline{L}$ denotes its complement. We identify an interpretation with the set of literals true in it.

By a *causal law* we mean an expression of the form

$$\phi \Rightarrow \psi \tag{1}$$

where ϕ and ψ are formulas of the underlying propositional language. By the *antecedent* and *consequent* of (1), we mean the formulas ϕ and ψ, respectively. We emphasize that (1) is not the material conditional $\phi \supset \psi$. The intended reading of (1) is: *Necessarily, if ϕ then the fact that ψ is caused.* Thus, we may say that (1) describes a condition under which ψ is caused.

By a *causal theory* we mean a set of causal laws.

2.2 SEMANTICS

For every causal theory D and interpretation I, let

$$D^I = \{\, \psi \,:\, \text{for some } \phi,\ \phi \Rightarrow \psi \in D \text{ and } I \models \phi \,\}.$$

Main definition. Let D be a causal theory. An interpretation I is *causally explained* according to D if I is the unique model of D^I.

As discussed in [McCain and Turner, 1997], this definition reflects a principle of "universal causation": every literal that is true in a causally explained interpretation I according to D is required to be caused in I according to D. Intuitively, when D describes an action domain, the interpretations that are causally explained according to D correspond to the world histories that are causally possible according to D. For a more helpful introduction to the causal theories formalism, see [McCain and Turner, 1997].

2.3 LITERAL COMPLETION

A causal theory D is *definite* if

- the consequent of each causal law in D is either a literal or *False*, and
- every literal is the consequent of finitely many causal laws in D.

Notice that, due to the first condition, an interpretation I is causally explained according to a definite causal theory D if and only if $I = D^I$.

In this paper, we will be particularly interested in definite causal theories, because they have a concise translation into classical propositional logic.

Let D be a definite causal theory. By the *literal completion* of D, denoted by *lcomp*(D), we mean the classical propositional theory obtained by an elaboration of the Clark completion method [Clark, 1978], as follows. For each literal L in the language of D, include in *lcomp*(D) the formula

$$L \equiv (\phi_1 \vee \cdots \vee \phi_n) \tag{2}$$

where $\phi_1, \ldots, \phi_n$ are the antecedents of the causal laws in D with consequent L. (Of course, if no causal law in D has consequent L, then (2) becomes $L \equiv \mathit{False}$.) We will call formula (2) the *completion* of L. Also, for each causal law of the form $\phi \Rightarrow \mathit{False}$ in D, include in *lcomp*(D) the formula $\neg\phi$. We will sometimes refer to causal laws with consequent *False* as *constraints*.

For example, let D_1 be the causal theory (in the language with exactly the atoms p and q) consisting of the causal laws

$$p \Rightarrow p\,,\ \neg q \Rightarrow p\,,\ q \Rightarrow q\,,\ \neg q \Rightarrow \neg q\,,\ q \Rightarrow \mathit{False}\,.$$

Causal theory D_1 is definite, and *lcomp*(D_1) is

$$\{\, p \equiv p \vee \neg q\,,\ \neg p \equiv \mathit{False}\,,\ q \equiv q\,,\ \neg q \equiv \neg q\,,\ \neg q \,\}\,.$$

The following proposition generalizes slightly a result presented without proof in [McCain and Turner, 1997].

Proposition 1 *An intepretation I is causally explained according to a definite causal theory D if and only if I is a model of lcomp(D).*

Proof. Assume $I = D^I$. Then for every literal $L \in I$, (i) there is a formula ϕ such that $\phi \Rightarrow L$ belongs to D and $I \models \phi$, and (ii) there is no formula ϕ such that $\phi \Rightarrow \overline{L}$ belongs to D and $I \models \phi$. It follows that for every literal $L \in I$, (i) I satisfies the completion of L, and (ii) I satisfies the completion of $\overline{L}$. That is, I satisfies the completion of every literal in the language of D. Similarly, since $\mathit{False} \notin D^I$, we can conclude that I satisfies every formula in *lcomp*(D) obtained from a constraint. So I is a model of *lcomp*(D). Proof in the other direction is similar. □

Let D be a causal theory, Γ a set of formulas, and ϕ a formula. We write

$$\Gamma \vdash_D \phi \tag{3}$$

to say that ϕ is true in every model of Γ that is causally explained according to D.

The following corollary suggests an approach to query evaluation for definite causal theories.

Corollary 1 *Let D be a definite causal theory, Γ a set of formulas, and ϕ a formula. $\Gamma \vdash_D \phi$ if and only if lcomp$(D) \cup \Gamma \cup \{\neg\phi\}$ is unsatisfiable.*

3 CAUSAL ACTION THEORIES

3.1 $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ LANGUAGES

When representing an action domain by a causal theory, it is convenient to describe the underlying propositional signature by means of three pairwise-disjoint sets: a nonempty set $\mathbf{F}$ of *fluent names*, a set $\mathbf{A}$ of *action names*, and a nonempty set $\mathbf{T}$ of *time names* (corresponding to the natural numbers or an initial segment of the natural numbers). The atoms of the language $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ are divided into two classes, defined as follows. The *fluent atoms* are expressions of the form f_t such that $f \in \mathbf{F}$ and $t \in \mathbf{T}$. Intuitively, f_t is true if and only if the fluent f holds at time t. The *action atoms* are expressions of the form a_t such that $a \in \mathbf{A}$ and $t, t{+}1 \in \mathbf{T}$.[2] Intuitively, a_t is true if and only if the action a occurs at time t.

An *action literal* is an action atom or its negation. A *fluent literal* is a fluent atom or its negation. A *fluent formula* is a propositional combination of fluent atoms. We say that a formula refers to a time t if an atom of the form x_t occurs in it.

An $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ *domain description* is a causal theory in an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ language.

3.2 EXAMPLE DOMAIN DESCRIPTIONS

To illustrate our approach to action formalization, we will represent a "falling dominos" domain and a "pendulum" domain. (For a gentler introduction, see [McCain and Turner, 1997].) These examples demonstrate interesting expressive possibilities of causal theories. Both descriptions are definite, and, as we will see, suitable for satisfiability planning.

3.2.1 Dominos Domain

We wish to describe the chain reaction of dominos falling over one after the other, after the first domino is tipped over.

Let the fluent names be $\mathit{Up}(1), \ldots, \mathit{Up}(4)$, and let the single action name be *Tip*. Identify time with the natural numbers $0, \ldots, 4$.

Here, as usual, we assume that facts about the occurrences of actions are *exogenous* to the causal theory. We express this assumption by writing the following schemas, where A is a meta-variable for action names. (Throughout, t is a meta-variable for time names.)

$$A_t \Rightarrow A_t \tag{4}$$

$$\neg A_t \Rightarrow \neg A_t \tag{5}$$

[2] As you might expect, the expression $t{+}1$ stands for the name of the successor of the number named by t.

Schema (4) says that the occurrence of an action A at a time t is caused whenever A occurs at t. Similarly, schema (5) says that the non-occurrence of A at time t is caused whenever A does not occur at t.

We also typically assume that facts about the initial values of fluents are exogenous, by writing the schemas

$$F_0 \Rightarrow F_0 \tag{6}$$
$$\neg F_0 \Rightarrow \neg F_0 \tag{7}$$

where F is a meta-variable for fluent names.

The fluent names $Up(1), \ldots, Up(4)$ will be designated *inertial*. For inertial fluents we write the following schemas, where I is a meta-variable for inertial fluent names.

$$I_t \wedge I_{t+1} \Rightarrow I_{t+1} \tag{8}$$
$$\neg I_t \wedge \neg I_{t+1} \Rightarrow \neg I_{t+1} \tag{9}$$

The first schema says that whenever an inertial fluent remains true from one time to the next, its truth at the latter time is caused. The second schema is similar. Taken together, these schemas solve the frame problem for inertial fluents.

We describe the direct effect and action precondition of the *Tip* action by writing

$$Tip_t \Rightarrow \neg Up(1)_{t+1} \tag{10}$$
$$\neg Up(1)_t \wedge Tip_t \Rightarrow False\,. \tag{11}$$

So *Tip* is the action of tipping over the first domino. It can only be done if the first domino is standing up.

We describe the chain reaction mechanism as follows, where d is a meta-variable ranging over numbers $1, 2, 3$.

$$Up(d)_t \wedge \neg Up(d)_{t+1} \Rightarrow \neg Up(d+1)_{t+2} \tag{12}$$

Notice that this schema does not mention an action. It describes dynamic change involving three distinct time points. Roughly speaking, if domino d falls in the interval from t to $t+1$, then domino $d+1$ is caused to fall in the interval from $t+1$ to $t+2$.

Let D_2 be the causal theory given by schemas (4)–(12). Let I be the interpretation shown below.

Tip_0	$\neg Tip_1$	$\neg Tip_2$	$\neg Tip_3$	
$Up(1)_0$	$\bullet\ \neg Up(1)_1$	$\neg Up(1)_2$	$\neg Up(1)_3$	$\neg Up(1)_4$
$Up(2)_0$	$Up(2)_1$	$\bullet\ \neg Up(2)_2$	$\neg Up(2)_3$	$\neg Up(2)_4$
$Up(3)_0$	$Up(3)_1$	$Up(3)_2$	$\bullet\ \neg Up(3)_3$	$\neg Up(3)_4$
$Up(4)_0$	$Up(4)_1$	$Up(4)_2$	$Up(4)_3$	$\bullet\ \neg Up(4)_4$

Interpretation I specifies, for all actions a and times t such that $t+1 \in \mathbf{T}$, whether or not a occurs at t, and, for all fluents f and times t, whether or not f holds at t. The only action occurrence is *Tip* at time 0. One easily verifies that $I = D_2^I$. The four literals preceded by bullets appear in D_2^I due to the domain specific schemas (10) and (12). The other literals appear due to "standard schemas" (4)–(9). Hence, I is causally explained according to D_2.

3.2.2 Pendulum Domain

As a second example, we will formalize a dynamic domain from [Giunchiglia and Lifschitz, 1998]. In this domain there is a pendulum. In the course of nature (i.e., in the absence of interventions), the pendulum bob swings back and forth from right to left. However, at any time the agent can intervene to change the course of nature by holding the bob in its current location. So long as he continues to hold it, the bob remains where it is. When he no longer holds it, the bob resumes its natural course, swinging back and forth from right to left.

In formalizing the Pendulum domain, we will use a single action name *Hold* and fluent name *Right*. We will identify time with the natural numbers $0, \ldots, 4$.

The effects of the action *Hold* are specified straightforwardly by writing

$$Hold_t \wedge Right_t \Rightarrow Right_{t+1} \tag{13}$$
$$Hold_t \wedge \neg Right_t \Rightarrow \neg Right_{t+1}. \tag{14}$$

The behavior of the pendulum in the absence of interventions is described by writing

$$\neg Right_t \wedge Right_{t+1} \Rightarrow Right_{t+1} \tag{15}$$
$$Right_t \wedge \neg Right_{t+1} \Rightarrow \neg Right_{t+1}. \tag{16}$$

Like schemas (8) and (9) for inertia, schemas (15) and (16) describe a course of nature. Here the course of nature is dynamic rather than static, but otherwise there are clear similarities between the two pairs of schemas. Both pairs allow for the possibility that the course of nature may be overridden by the effects of actions, and both achieve this without mentioning facts about the non-occurrence of actions as preconditions.

In essence, schemas (15) and (16) solve the frame problem for the dynamic fluent *Right* in the same way that (8) and (9) solve the frame problem for inertial fluents.

Let D_3 be the causal theory given by schemas (4)–(7) and (13)–(16). Let the interpretation I be as follows.

$\neg Hold_0$	$Hold_1$	$Hold_2$	$\neg Hold_3$	
$Right_0$	$\neg Right_1$	$\neg Right_2$	$\neg Right_3$	$Right_4$

One easily verifies that $I = D_3^I$. Hence, I is causally explained according to D_3.

4 PLANNING WITH CAUSAL ACTION THEORIES

In this section we define fundamental notions related to planning, in the setting of causal action theories.

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description. By an *initial state description* we mean a set S_0 of fluent literals that refer to time 0 such that (1) for every fluent name $F \in \mathbf{F}$, exactly one of $F_0, \neg F_0$ belongs to S_0, and (2) $S_0 \not\vdash_D \mathit{False}$. Intuitively, an initial state description specifies an initial state that occurs in some causally possible world, i.e., a causally possible initial state. By a *time-specific goal*, we simply mean a fluent formula. Notice that a time-specific goal may refer to more than one time. By an *action history* we mean a set P of action literals such that, for every action name $A \in \mathbf{A}$ and time t such that $t+1 \in \mathbf{T}$, exactly one of $A_t, \neg A_t$ belongs to P. Every interpretation includes exactly one action history.

We will define when an action history P is a valid plan for achieving a time-specific goal G in an initial state S_0. This definition rests on the more fundamental notions of sufficiency and executability, which we also define. We define two other properties of plans, more naturally associated with satisfiability planning. One is determinism. The other is discussed next.

4.1 CAUSALLY POSSIBLE PLANS

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. An action history P is a *causally possible* plan for achieving G in S_0 if

$$S_0 \cup P \not\vdash_D \neg G\,.$$

This condition says that there is, intuitively speaking, some causally possible world in which G can be achieved by executing P in initial state S_0.

Corollary 1 yields the following proposition.

Proposition 2 *Let D be a definite $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. An action history P is included in a model of $\mathit{lcomp}(D) \cup S_0 \cup \{G\}$ if and only if P is a causally possible plan for achieving G in S_0.*

This proposition guarantees that every plan obtained by the satisfiability method is causally possible. Unfortunately, this is a rather weak guarantee. For example, in a nondeterministic "coin tossing" domain, a causally possible plan for having the coin lie heads at time 1, after lying tails at time 0, is simply to toss the coin at time 0. We can make this precise, as follows. We use a single fluent name *Heads*, a single action name *Toss*, and two times, 0 and 1. We designate *Heads* inertial. The causal theory D_4 for the action domain is represented by standard schemas (4)–(9), along with domain specific schemas (17) and (18) below.

$$\mathit{Toss}_t \wedge \mathit{Heads}_{t+1} \Rightarrow \mathit{Heads}_{t+1} \tag{17}$$

$$\mathit{Toss}_t \wedge \neg\mathit{Heads}_{t+1} \Rightarrow \neg\mathit{Heads}_{t+1} \tag{18}$$

Take $S_0 = \{\neg\mathit{Heads}_0\}$, $G = \mathit{Heads}_1$, and $P = \{\mathit{Toss}_0\}$. One easily checks that the interpretation $S_0 \cup P \cup \{G\}$ is causally explained according to D_4. Hence, P is a causally possible plan for achieving Heads_1 in S_0. On the other hand, P is also a causally possible plan for achieving $\neg\mathit{Heads}_1$ in S_0, since $S_0 \cup P \cup \{\neg G\}$ is also causally explained according to D_4.

4.2 SUFFICIENT PLANS

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. An action history P is a *sufficient* plan for achieving G in S_0 if

$$S_0 \cup P \vdash_D G\,.$$

Intuitively, according to this definition, G will be achieved whenever P is done starting in S_0.

Sufficiency does not say anything about whether P can be executed in S_0, so it is not surprising that some sufficient plans are not valid. In fact, even plans that are both causally possible and sufficient can fail to be valid. Here is an example, again involving coin tossing, along with a second action of truly saying that the coin lies heads. We have a single fluent name, *Heads*, two action names, *Toss* and *TrulySayHeads*, and three times, 0, 1 and 2. Again *Heads* is inertial. The causal theory D_5 for this action domain is represented by standard schemas (4)–(9), together with domain specific schemas (17) and (18) from D_4, and one additional domain specific schema, shown below.

$$\mathit{TrulySayHeads}_t \wedge \neg\mathit{Heads}_t \Rightarrow \mathit{False} \tag{19}$$

Take $S_0 = \{\neg\mathit{Heads}_0\}$, $G = \mathit{Heads}_2$, and let P consist of $\mathit{Toss}_0, \neg\mathit{TrulySayHeads}_0, \neg\mathit{Toss}_1, \mathit{TrulySayHeads}_1$. So the plan is to toss the coin and then truly say heads. There is exactly one model of $S_0 \cup P$ that is causally explained by D_5—namely, the interpretation $S_0 \cup P \cup \{\mathit{Heads}_1, \mathit{Heads}_2\}$. Therefore, P is a sufficient, causally possible plan for achieving Heads_2 in S_0. That is, roughly speaking, there is a causally possible world in which doing P in S_0 achieves Heads_2, and, moreover, in any causally possible world in which P is done in S_0, Heads_2 is achieved. Nonetheless, P is not a valid plan. Intuitively, the problem is that P is not executable in S_0—it could be that the coin comes

up tails after the initial toss, in which case the agent cannot truly say heads at time 1.

4.3 EXECUTABLE PLANS

We next define when a plan is executable in an initial state. Unfortunately, this condition is less convenient to state and check than the previous ones.

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description. For any time name $t \in \mathbf{T}$, let $\mathbf{T}|t = \{s \in \mathbf{T} : s \leq t\}$. Given a set X of $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ literals, and a time name t, we write $X|t$ to denote the set of all literals in X that belong to the restricted language $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T}|t)$.

Let P be an action history and S_0 an initial state description. We specify when $P|t$ is executable in S_0 by the following recursive definition. $P|0$ is *executable* in S_0. (Note that $P|0 = \emptyset$.) For all times $t+1 \in \mathbf{T}$, $P|t+1$ is *executable* in S_0 if the following two conditions hold: (i) $P|t$ is executable in S_0, and (ii) for every causally explained interpretation I that satisfies $S_0 \cup P|t$, there is a model of $I|t \cup P|t+1$ that is causally explained. Finally, we say that P itself is *executable* in S_0 if, for every time $t \in \mathbf{T}$, $P|t$ is executable in S_0.

So a plan P is executable if all of its prefixes are. Recall that a prefix $P|t$ completely specifies all action occurrences before time t, and that $P|t+1$ specifies in addition the action occurrences at time t. Thus $P|t+1$ is executable, roughly speaking, if $P|t$ is and, no matter the state of the world after executing $P|t$, the actions specified by P for time t can then be performed.

For example, consider more closely why the plan P from the last example is not executable in S_0. Recall that initially the coin lies tails. The prefix $P|1$ is executable in S_0. That is, it is possible to toss and not truly say heads at time 0. But prefix $P|2$ is not executable in S_0. Intuitively, it may not be possible to truly say heads at time 1. More precisely, notice that the interpretation I obtained from $S_0 \cup P|1$ by adding $\neg \mathit{Heads}_1, \neg \mathit{Toss}_1, \neg \mathit{TrulySayHeads}_1, \neg \mathit{Heads}_2$ is causally explained according to D_5, yet no model of $I|1 \cup P|2$ is causally explained. This is because no causally explained interpretation satisfies both $\neg \mathit{Heads}_1$ and $\mathit{TrulySayHeads}_1$.

4.4 VALID PLANS

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. An action history P is a *valid* plan for achieving G in S_0 if it is both sufficient and executable.

The next proposition shows that valid plans are causally possible.

Lemma 1 *Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description and S_0 an initial state description. If an action history P is executable in S_0, then there is model of $S_0 \cup P$ that is causally explained according to D.*

Proof Sketch. The definition of the executability of P in S_0 provides a basis for constructing a causally explained interpretation I such that, for all times $t \in \mathbf{T}$, I satisfies $S_0 \cup P|t$. □

Proposition 3 *Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. If P is a valid plan for achieving G in S_0, then it is a causally possible plan for achieving G in S_0.*

Proof. By Lemma 1, since P is executable in S_0, some model I of $S_0 \cup P$ is causally explained. Since P is sufficient for G in S_0, $S_0 \cup P \vdash_D G$. Hence, I satisfies G, which shows that $S_0 \cup P \nvdash_D \neg G$. □

4.5 DETERMINISTIC PLANS

We will define one more class of plans, the deterministic plans. We will show that if a plan is causally possible and deterministic, it is valid. This is a key result in our approach to satisfiability planning. In Section 5 we will introduce the class of simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain descriptions, and show that for them all causally possible plans are deterministic, and thus valid.

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, P an action history, and S_0 an initial state description. For every time t, $P|t$ is *deterministic* in S_0 if for all fluent names F and times $s \leq t$, $S_0 \cup P|t \vdash_D F_s$ or $S_0 \cup P|t \vdash_D \neg F_s$. We say that P is *deterministic* in S_0 if for every time $t \in \mathbf{T}$, $P|t$ is deterministic in S_0.

Thus a plan P is deterministic if all of its prefixes are. Recall that a prefix $P|t$ is a complete specification of action occurrences for all times before t. Prefix $P|t$ is deterministic if, roughly speaking, performance of the actions in $P|t$ starting in S_0 would completely determine the values of all fluents up to time t.

This definition yields a strong lemma.

Lemma 2 *Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description and S_0 an initial state description. If an action history P is deterministic in S_0, then at most one model of $S_0 \cup P$ is causally explained according to D.*

Proof. Let I and I' be causally explained models of $S_0 \cup P$. Consider any fluent atom F_t. Since P is deterministic in S_0, so is $P|t$. Since both I and I' satisfy $S_0 \cup P|t$, it follows that they agree on F_t. Hence $I = I'$. □

The converse of Lemma 2 does not hold. P may fail

to be deterministic in S_0 even when there is at most one causally explained model of $S_0 \cup P$. We illustrate this with another coin tossing example. Take a single fluent name, *Heads*, and three action names, *Toss*, *TrulySayHeads* and *TrulySayTails*. Identify time with the natural numbers. Once more we designate *Heads* inertial. The domain description D_6 is represented by standard schemas (4)–(9), together with domain specific schemas (17), (18) and (19) from D_5, and two more domain specific schemas, shown below.

$$TrulySayTails_t \wedge Heads_t \Rightarrow False \quad (20)$$
$$TrulySayHeads_t \equiv TrulySayTails_t \Rightarrow False \quad (21)$$

Due to (21), exactly one non-toss action occurs at every time in every causally possible world. Moreover, by (19) and (20), whenever truly say heads occurs, the coin lies heads, and whenever truly say tails occurs, the coin lies tails. Thus, each causally possible world is completely determined by its initial state and the actions that are performed in it. For instance, let $S_0 = \{\neg Heads_0\}$ and consider the plan P in which the agent initially tosses and concurrently truly says tails, and forever after truly says heads and does not toss. Although exactly one model of $S_0 \cup P$ is causally explained, P is not deterministic. This is because $P|1$ is not deterministic. That is, tossing and concurrently truly saying tails at time 0 simply does not determine the state of the coin at time 1.

Proposition 4 *Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. If P is a causally possible plan for achieving G in S_0 and P is also deterministic in S_0, then P is a valid plan for achieving G in S_0.*

Proof. Since P is a causally possible plan for achieving G in S_0, some model I^* of $S_0 \cup P \cup \{G\}$ is causally explained. By Lemma 2, no other model of $S_0 \cup P$ is causally explained. Since I^* satisfies G, P is sufficient for achieving G in S_0. To show that P is executable in S_0, we prove by induction that for all times t, $P|t$ is executable in S_0. The base case is trivial. For the inductive step, we show that $P|t+1$ is executable in S_0. By the inductive hypothesis, $P|t$ is executable in S_0. Thus we can complete the proof as follows. Assume that I is a causally explained model of $S_0 \cup P|t$. Notice that both I and I^* satisfy $S_0 \cup P|t$. Since P is deterministic in S_0, so is $P|t$, and it follows that $I^*|t = I|t$. Since I^* also satisfies $P|t+1$, we're done. □

Of course the converse of Proposition 4 does not hold, since valid plans need not be deterministic.

5 SATISFIABILITY PLANNING WITH CAUSAL THEORIES

In this section, we consider how to restrict definite $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain descriptions so that the causally possible plans are deterministic and thus, by Proposition 4, valid. To this end, we introduce the class of "simple" $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain descriptions.

5.1 SIMPLE DOMAIN DESCRIPTIONS

A definite $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description D is *simple* if it has the following three (yet to be defined) properties: it is inertially unambiguous, adequately acyclic, and respects the flow of time.

5.1.1 Inertially Unambiguous

Let $\mathbf{F}^+$ denote the set of all fluent atoms that refer to nonzero times. Causal laws in D of the form $\phi \wedge L \Rightarrow L$, where $L \in \mathbf{F}^+$ or $\overline{L} \in \mathbf{F}^+$, and ϕ is any $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ formula, will be called *inertia-like laws*.

Note that this definition covers not only causal laws obtained from standard inertia schemas (8)–(9) but also, for instance, causal laws such as those obtained from schemas (15)–(16) in the Pendulum domain D_3, which describe a dynamic course of nature. This definition also covers causal laws such as those obtained from schemas (17)–(18) in the coin-tossing domains. Although these causal laws express the direct nondeterministic effect of the coin-tossing action, they have a form similar to that of inertia laws.

We say that D is *inertially unambiguous* if it includes no pair of inertia-like laws

$$\phi \wedge F_{t+1} \Rightarrow F_{t+1} \quad (22)$$
$$\psi \wedge \neg F_{t+1} \Rightarrow \neg F_{t+1} \quad (23)$$

such that the formula $\phi \wedge \psi$ is satisfiable.

This exclusivity condition on ϕ and ψ is the only non-syntactic component of the definition of a simple domain description. Notice that the pairs of laws represented by schemas (8)–(9) for inertia and schemas (15)–(16) in the Pendulum domain satisfy this condition.

5.1.2 Adequately Acyclic

The *proper atom dependency graph* of D is the directed graph defined as follows. Its nodes are the atoms of the language of D. Let D' be the causal theory obtained from D by (i) deleting all causal laws whose consequent is *False*, and (ii) replacing each inertia-like law $\phi \wedge L \Rightarrow L$ with the causal law $\phi \Rightarrow L$. For each

causal law in D', there is an edge from the atom that occurs in the consequent to each atom that occurs in the antecedent. We use the proper atom dependency graph to define an ordering on $\mathbf{F}^+$ as follows. For all $A, A' \in \mathbf{F}^+$, $A <_D A'$ if there is a nonempty path from A' to A. (So the edges in the graph point downward in the ordering.) We say that D is *adequately acyclic* if the ordering $<_D$ on $\mathbf{F}^+$ is well-founded.

Intuitively, this condition restricts cyclic causal dependencies between fluents, while allowing cycles that arise due to causal laws related to inertia.

5.1.3 Respects the Flow of Time

We say that D *respects the flow of time* if every causal law in D satisfies the following two conditions.

- If the consequent refers to a time t, then the antecedent does not refer to a time later than t.
- If the consequent is a fluent literal that refers to time t, then every action atom in the antecedent refers to a time earlier than t.

Notice that the description D_2 of the Dominos domain is simple, as is the description D_3 of the Pendulum domain. The coin-tossing domains D_4, D_5 and D_6 are not, because they are not inertially unambiguous.

5.2 SIMPLE DOMAIN DESCRIPTIONS YIELD VALID PLANS

Here is the main technical result related to simple domain descriptions. Its proof is postponed to Section 8.

Proposition 5 *Let D be a simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. If P is a causally possible plan for achieving G in S_0, then P is a valid plan for achieving G in S_0.*

From this result, along with Propositions 2 and 3, we obtain the following characterization of satisfiability planning with simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain descriptions.

Theorem 1 *Let D be a simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and G a time-specific goal. An action history P is included in a model of*

$$lcomp(D) \cup S_0 \cup \{G\}$$

if and only if P is a valid plan for achieving G in S_0.

For effective satisfiability planning, we must of course also require that the simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description be finite, with $\mathbf{F}$, $\mathbf{A}$, and $\mathbf{T}$ finite as well.

6 SATISFIABILITY PLANNING PROGRAM

Given a finite signature and a set of schemas representing a finite, definite causal theory, it is straightforward to instantiate the schemas to obtain the represented (ground) causal theory, form its literal completion, and convert it to clausal form. We have written a Prolog program to carry out these tasks. It includes a procedure named `load_file/1`, which reads in a file such as the one displayed in Figure 1 for the Pendulum domain, and writes out in clausal form the literal completion of the causal theory. In the input syntax, fluent atoms f_t are represented as `h(`f`,`t`)`, and action atoms a_t are represented as `o(`a`,`t`)`. The symbols `h` and `o` are read as "holds" and "occurs," respectively.

```
% File: pendulum
:- declare_types  type(fluent,[right]),
  type(action,[hold]), type(time,[0..4]),
  type(atom,[h(fluent,time),o(action,time)]).
:- declare_variables  var(A,action),
  var(F,fluent), var([T,T1],time).
% domain specific schemas
  o(hold,T) & h(right,T) => h(right,T1)
                              where T1 is T+1.
  o(hold,T) & -h(right,T) => -h(right,T1)
                              where T1 is T+1.
  -h(right,T) & h(right,T1) => h(right,T1)
                              where T1 is T+1.
  h(right,T) & -h(right,T1) => -h(right,T1)
                              where T1 is T+1.
% standard schemas
  o(A,T) => o(A,T).  -o(A,T) => -o(A,T).
  h(F,0) => h(F,0).  -h(F,0) => -h(F,0).
```

Figure 1: Example Input File for the Planning System

After `load_file/1` has processed the domain description, planning problems are posed by calling the procedure `plan/0`, as shown in Figure 2. The procedure `plan/0` reads in an initial state description S_0 and a time-specific goal G, converts them to clausal form, and adds them to the clause set obtained from the domain description.[3] The resulting clause set is simplified, as in [Kautz and Selman, 1996][4], and submitted to the satisfiablity checker *rel_sat* [Bayardo and Schrag, 1997]. If $lcomp(D) \cup S_0 \cup \{G\}$ is satisfiable, *rel_sat* finds a satisfying interpretation and `plan/0` displays it, answering "yes." The plan P can be read off from this display. By Theorem 1, if D is simple, P is guaranteed to be a valid plan for achieving the goal G starting in initial state S_0. If *rel_sat* fails to find a sat-

[3] As illustrated in Figure 2, the initial state description S_0 can be replaced by a set Γ of formulas referring only to time 0 such that $\Gamma \vdash_D \phi$, where ϕ is the conjunction of the members of S_0, and yet $\Gamma \not\vdash_D False$.

[4] Steps: subsumption, unit propogation, subsumption.

isfying interpretation, `plan/0` answers "no." Since the solver *rel_sat* is systematic, we know in this case that G cannot be achieved starting in S_0. In either case, the time spent in the solver *rel_sat* is reported.

```
| ?- load_file(pendulum).
% 9 atoms, 28 rules, 16 clauses loaded.
yes
| ?- plan.
enter facts and goal (then ctrl-d)
|: h(right,0).
|: -h(right,2) & h(right,4).
|:
0.  right
Actions: hold
1.  right
Actions:
2.  -right
Actions: hold
3.  -right
Actions:
4.  right
Elapsed Time (cpu sec): 0.01
yes
```

Figure 2: Planning Session with Pendulum Domain

7 LARGE PLANNING PROBLEMS

Here we report on the performance of our approach when applied to the large blocks world and logistics planning problems from [Kautz and Selman, 1996]. As far as we know, the results obtained there compare favorably with the best current general-purpose planning systems. We obtain comparable results.

7.1 BLOCKS WORLD PROBLEMS

The four blocks worlds planning problems from [Kautz and Selman, 1996] are characterized below.

```
Blocks World A.  9 blocks.  Requires 6 moves.
   Initial state: 2/1/0  4/3  8/7/6/5
   Goal state:    4/0  7/8/3  1/2/6/5
Blocks World B. 11 blocks.  Requires 9 moves.
   Initial state: 2/1/0  10/9/4/3  8/7/6/5
   Goal state:    0/4/9  7/8/3  1/2/10/6/5
Blocks World C. 15 blocks.  Requires 14 moves.
   Initial: 2/1/0/11/12  10/9/4/3/13/14  8/7/6/5
   Goal:    13/0/4/9  14/12/7/8/3  11/1/2/10/6/5
Blocks World D. 19 blocks.  Requires 18 moves.
   0/11/12 10/9/4/3/13/14 8/7/6/5 18/17/16/15/2/1
   16/17/18/13/0/4/9 14/12/7/8/3 11/1/2/15/10/6/5
```

Our input file representing Blocks World D is displayed in Figure 3. We adapt the "operator splitting" approach used by Kautz and Selman. Instead of axiomatizing an action $Move(b, l', l)$, they axiomatize three "component" actions, which we can write: $Pickup(b)$, $Takefrom(l')$, $Putat(l)$. Their axioms are based on Schubert's "explanation closure" [1990], augmented with state constraints. In comparison, we introduce names for only two components of the move action: $Pickup(b)$, $Putat(l)$. (When moved, a block is taken from where it currently is.) We also do not introduce a fluent $Clear(b)$. Kautz and Selman include in their description a number of state constraints that we omit.[5] Preliminary experiments indicated that additional state constraints in our blocks world descriptions increase solution times on larger problems.

One can easily verify that the causal theory represented in Figure 3 is simple. The main complication, compared to our descriptions of the Dominos and Pendulum domains involves action atoms, which are largely irrelevant in determining whether a description is simple. Here we include a family of action atoms that are "true by default" rather than exogenous. Thus, for example, the action *NoPickup* is assumed to occur, roughly speaking, and we describe the conditions under which it is caused not to occur—whenever $PickUp(B)$ occurs, for some block B. These auxiliary "inaction" atoms are used to stipulate that a pickup action occurs if and only if a putat action does.

7.2 LOGISTICS PLANNING PROBLEMS

The logistics domain is due to Veloso [1992]. Kautz and Selman studied three large logistics planning problems. Due to space constraints, we do not describe them, nor do we display an example input file.

The logistics domain is more complex than the blocks world domain. It includes several kinds of actions that can occur concurrently. Our description of the logistics domain does not use operator splitting (which is not generally applicable to concurrent actions). Preliminary experiments indicated that, in constrast to the blocks world, logistics domain descriptions should include a variety of state constraints in order to get consistently good performance. We note that the logistics domain description used in our experiments is simple, and thus suitable for satisfiablity planning.

7.3 EXPERIMENTAL RESULTS

In our experimental results on these planning problems, we report the size of the clausal theory obtained from the literal completion of the causal action

[5] Their state constraints still do not rule out all "physically impossible" states. This is in accordance with the usual practice in describing action domains for planning. Roughly speaking, one need only say enough to guarantee that no "illegal" state can be reached from a legal one. Intuitively, this is adequate because planning problems are posed in part by specifying a legal initial state.

```
:- declare_types  type(block,[0..18]), type(location,[block,table]),
     type(fluent,[on(block,location)]), type(action,[pickup(block),putat(location)]),
     type(inaction,[nopickup,noputat]), type(time,[0..18]),
     type(atom,[o(action,time),o(inaction,time),h(fluent,time)]).
:- declare_variables  var([B,B1],block), var([L,L1],location),
     var(F,fluent), var(A,action), var(X,inaction), var([T,T1],time).
% state constraints: the first two allow concise input of initial state and goal
     h(on(B,L),0) & h(on(B,L1),0) => false  where B \== L, B \== L1, L @< L1.
     h(on(B,L),18) & h(on(B,L1),18) => false  where B \== L, B \== L1, L @< L1.
     h(on(B,B),T) => false.
% direct effects of actions
     o(pickup(B),T) & o(putat(L),T) => h(on(B,L),T1)  where T1 is T+1, B \== L.
     h(on(B,L),T) & o(pickup(B),T) => -h(on(B,L),T1)  where T1 is T+1, B \== L.
% explicit action preconditions
     o(pickup(B),T) & h(on(B1,B),T) => false  where B =\= B1.
     o(putat(B),T) & h(on(B1,B),T) => false  where B =\= B1.
     o(pickup(B),T) & o(putat(B),T) => false.
     o(pickup(B),T) & o(putat(table),T) & h(on(B,table),T) => false.
% at most one move action at a time
     o(pickup(B),T) & o(pickup(B1),T) => false  where B @< B1.
     o(putat(L),T) & o(putat(L1),T) => false  where L @< L1.
     o(pickup(B),T) => -o(nopickup,T).   o(putat(L),T) => -o(noputat,T).
     o(nopickup,T) & -o(noputat,T) => false.   -o(nopickup,T) & o(noputat,T) => false.
% standard schemas
     h(F,0) => h(F,0).   -h(F,0) => -h(F,0).
     h(F,T) & h(F,T1) => h(F,T1)  where T1 is T+1.
     -h(F,T) & -h(F,T1) => -h(F,T1)  where T1 is T+1.
     o(A,T) => o(A,T).   -o(A,T) => -o(A,T).   o(X,T) => o(X,T).
```

Figure 3: Input File for Blocks World D

Table 1: Satisfiability Planning with Causal Theories
Sizes are for clausal theories obtained, via literal completion, from causal action theories (after simplification). Time in seconds in solver *rel_sat* on Sparcstation 5.

Instance	Atoms	Clauses	Literals	Time
BW A	383	2412	5984	0.13
BW B	934	6241	15903	0.81
BW C	2678	18868	48704	35.2
BW D	5745	41726	108267	620.0
LOG A	1643	9205	20712	3.7
LOG B	1760	10746	24134	8.4
LOG C	2300	14450	32346	25.0

Table 2: Kautz and Selman Problem Descriptions
Here we establish the benchmarks—the results for the clausal theories used in [Kautz and Selman, 1996], with solution times obtained in the same manner as in Table 1.

Instance	Atoms	Clauses	Literals	Time
BW A	459	4675	10809	0.20
BW B	1087	13772	31767	1.4
BW C	3016	50457	114314	66.3
BW D	6325	131973	294118	1052.0
LOG A	1782	20895	42497	2.5
LOG B	2069	29508	59896	9.8
LOG C	2809	48920	99090	32.3

theory—in terms of numbers of atoms, clauses and literal occurrences, after simplification—and time spent in the solver, following the reporting methodology of [Kautz and Selman, 1996]. Solution times are averaged over 20 runs of the solver *rel_sat* on a Sparcstation 5, using different random number seeds. Table 1 displays statistics for finding plans by our method.

For the sake of comparison, we performed the corresponding experiments on the problem descriptions from [Kautz and Selman, 1996], again using the solver *rel_sat* on a Sparcstation 5.[6] The results appear in Table 2. Bayardo and Schrag [1997] showed that, for the clausal theories of Kautz and Selman that we consider,

[6]Kautz and Selman considered two kinds of descriptions of the logistics domains (both in classical propositional logic): one based on intuitions underlying the planner Graphplan [Blum and Furst, 1995]; the other obtained by first describing the domain as in explanation closure, then eliminating all action atoms. In this second case, a satisfying interpretation does not include an action history. Rather it provides, as it were, a refinement of the planning problem. That is, the satisfying interpretation can be understood as an initial state and goal which together specify completely the values of all fluent atoms. In our reported results, we refer to the first kind of description. We note in comparison that the solver *rel_sat* takes longer for each instance of the second kind of description.

their solver *rel_sat* outperforms both of the solvers—one systematic, one stochastic—used in [Kautz and Selman, 1996].[7] Notice that in all cases except Logistics A our solution times are better.

Finally, in order to show that a plan is optimal (in the number of time steps), it is necessary to show that no shorter plan exists. For this purpose it is essential that a systematic solver be used. In Table 3, we report on the performance of our approach for this task, again using the solver *rel_sat*. For each problem, we report the time to fail to find a plan one step shorter than the optimal plan. Notice that, for these planning problems, the time needed to fail is comparable to the time needed to succeed.

Table 3: Proving Plans Optimal: Satisfiability Planning with Causal Theories. Here, in each case, the domain description includes one time step less than needed for a solution. Time reported is number of seconds required for solver *rel_sat* to determine unsatisfiability.

Instance	Atoms	Clauses	Literals	Time
BW A	281	1741	4211	0.04
BW B	788	5246	13276	0.43
BW C	2420	17033	43865	21.6
BW D	5343	38795	100544	374.2
LOG A	1354	7378	16595	2.2
LOG B	1498	8908	20026	31.3
LOG C	1946	11924	26710	54.8

8 PROOF OF PROPOSITION 5

We begin with the main lemma.

Lemma 3 *Let D be a definite $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description that is inertially unambiguous and adequately acyclic. Let P be an action history and S_0 an initial state description. At most one model of $S_0 \cup P$ is causally explained according to D.*

Proof. We proceed by the method of contradiction. Suppose that two distinct causally explained interpretations I and I' satisfy $S_0 \cup P$. Let X be the set of atoms on which I and I' disagree. Notice that X is a nonempty subset of $\mathbf{F}^+$, since I and I' differ, and yet agree on all atoms not in $\mathbf{F}^+$. Let X' consist of the members of X that are minimal (among members of X) with respect to the ordering $<_D$. Notice that X' is nonempty, since X is nonempty and $<_D$ restricted to X is well-founded. Finally, let F_{t+1} be a member of X' whose time subscript is minimal (among members of X'). Without loss of generality, assume that $I \models F_{t+1}$ and $I' \models \neg F_{t+1}$. Since $I = D^I$ and $I' = D^{I'}$, there must be a pair of causal laws $\phi \Rightarrow F_{t+1}$ and $\psi \Rightarrow \neg F_{t+1}$ in D such that $I \models \phi$ but $I' \not\models \phi$, and $I \not\models \psi$ but $I' \models \psi$. It follows that I and I' differ on at least one atom A that occurs in ϕ. Thus, $A \in X$ and also $A <_D F_{t+1}$. Consequently, by the minimality of F_{t+1}, A is F_{t+1}. Since D is adequately acyclic, $\phi \Rightarrow F_{t+1}$ must be of the form (22), and so can be written $\phi' \wedge F_{t+1} \Rightarrow F_{t+1}$. Since $I \models \phi$, $I \models \phi'$. A similar argument shows that $\psi \Rightarrow \neg F_{t+1}$ has form (23), and can be written $\psi' \wedge \neg F_{t+1} \Rightarrow \neg F_{t+1}$, with $I' \models \psi'$. Because D is inertially unambiguous, I' cannot satisfy both ϕ' and ψ'. Hence $I' \not\models \phi'$. (We complete the proof by showing that $I' \models \phi'$.) We have already shown that the only atom in ϕ on which I and I' differ is F_{t+1}, which is to say that the only atom in $\phi' \wedge F_{t+1}$ on which I and I' differ is F_{t+1}. Since D is adequately acylic, we know F_{t+1} does not occur in ϕ'. So I and I' agree on all atoms in ϕ', and since $I \models \phi'$, $I' \models \phi'$ as well. Contradiction. □

Let D be an $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description. For any $t \in \mathbf{T}$, let $D|t$ be the causal theory in the restricted language $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T}|t)$ consisting of all causal laws from D in that language.

Observe that if D is a simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, then, for every time t, $D|t$ is a simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T}|t)$ domain description.

Lemma 4 *Let D be a simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description and P an action history. For all $t \in \mathbf{T}$, if I is a model of $S_0 \cup P|t$ that is causally explained according to D, then $I|t$ is the unique model of $S_0 \cup P|t$ causally explained according to $D|t$.*

Proof. Clearly $I|t$ is a model of $S_0 \cup P|t$. Given that D respects the flow of time, one easily verifies that $(D|t)^{I|t} = D^I|t$. Since $I = D^I$, $I|t = D^I|t$. So $I|t = (D|t)^{I|t}$, and we've shown that $I|t$ is a model of $S_0 \cup P|t$ that is causally explained according to $D|t$. We know that $D|t$ is simple since D is, so we can conclude by Lemma 3 that $I|t$ is unique. □

Lemma 5 *Let D be a simple $\mathcal{L}(\mathbf{F},\mathbf{A},\mathbf{T})$ domain description, S_0 an initial state description, and P an action history. If D has a causally explained interpretation satisfying $S_0 \cup P$, then P is deterministic in S_0.*

Proof. We need to show that, for all times $t \in \mathbf{T}$, $P|t$ is deterministic in S_0. Proof is by induction on t. The base case is trivial. By the inductive hypothesis, $P|t$ is deterministic in S_0. Assume that I and I' are models

[7] On the other hand, for their description of the logistics domain in which the action names are eliminated, their stochastic solver (properly tuned) is faster than *rel_sat*.

of $S_0 \cup P|t+1$ that are causally explained according to D. We need to show that $I|t+1 = I'|t+1$, which follows easily from Lemma 4. □

Proposition 4 and Lemma 5 yield Proposition 5.

9 CONCLUSION

This paper provides a theoretical foundation for satisfiability planning with causal theories. In our approach, action domain descriptions expressed as causal theories are translated into classical propositional logic. As we show, the classical models of the translation correspond exactly to the "causally possible" world histories according to the causal theory. Following Kautz and Selman, we then find plans by extracting them from models obtained by satisfiability checking.

In order to establish a basis upon which to judge the soundness of this approach to planning, we define a family of fundamental properties a plan may have: causally possible, deterministic, sufficient, executable. A plan is valid if and only if it is sufficient and executable. We observe that the plans obtained by the satisfiability method may, in general, fail to be sufficient or executable. They are only guaranteed to be causally possible. We show though that any causally possible plan that is deterministic is also valid.

We identify a class of "simple" domain descriptions for which the satisfiability method is applicable. Simple domain descriptions have a concise translation into classical logic. Moreover, we show that for such domains, the causally possible plans are deterministic and thus valid.

We describe an implemented satisfiability planning system based on these ideas, and provide experimental evidence that the approach can be computationally effective, by solving hard classical planning instances from [Kautz and Selman, 1996] comparatively quickly.

These developments are particularly noteworthy because of the expressive potential of simple causal theories, as illustrated by the Dominos and Pendulum domains presented in this paper. Thus, future applications of satisfiability planning with causal theories may address extensions to classical planning involving such features as concurrent actions and dynamic worlds.

Acknowledgments

We are grateful to our advisor, Vladimir Lifschitz. Thanks also to Enrico Giunchiglia. This work partially supported by NSF grant IRI-9306751. In addition, Norman McCain was partially supported by Texas Advanced Research Program grant 003658-242.

References

[Bayardo and Schrag, 1997] Roberto Bayardo and Robert Schrag. Using CSP look-back techniques to solve real-world SAT instances. In *Proc. of AAAI-97*, pages 203–208, 1997.

[Blum and Furst, 1995] A. Blum and M.L. Furst. Fast planning through planning graph analysis. In *Proc. IJCAI-95*, pages 1636–1642, 1995.

[Clark, 1978] Keith Clark. Negation as failure. In Herve Gallaire and Jack Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, 1978.

[Giunchiglia and Lifschitz, 1998] Enrico Giunchiglia and Vladimir Lifschitz. An action language based on causal logic. In Working Notes of 4th Symp. on Logical Formalizations of Commonsense Reasoning, 1998.

[Kautz and Selman, 1992] Henry Kautz and Bart Selman. Planning as satisfiability. In *Proc. of ECAI-92*, pages 359–379, 1992.

[Kautz and Selman, 1996] Henry Kautz and Bart Selman. Pushing the envelope: planning, propositional logic, and stochastic search. In *Proceedings of AAAI-96*, pages 1194–1201, 1996.

[Lifschitz, 1997] Vladimir Lifschitz. On the logic of causal explanation. *Artificial Intelligence*, 96:451–465, 1997.

[McCain, 1997] Norman McCain. *Causality in Commonsense Reasoning about Actions.* PhD thesis, UT Austin, Dept. of Computer Sciences, 1997.

[McCain and Turner, 1995] Norman McCain and Hudson Turner. A causal theory of ramifications and qualifications. In *Proc. of IJCAI-95*, pages 1978–1984, 1995.

[McCain and Turner, 1997] Norman McCain and Hudson Turner. Causal theories of action and change. In *Proc. of AAAI-97*, pages 460–465, 1997.

[Schubert, 1990] Lenhart Schubert. Monotonic solution of the frame problem in the situation calculus: an efficient method for worlds with fully specified actions. In H.E. Kyburg, R. Loui, and G. Carlson, editors, *Knowledge Representation and Defeasible Reasoning*, pages 23–67. Kluwer, 1990.

[Turner, 1998] Hudson Turner. A logic of universal causation. To appear in *Artificial Intelligence*, 1998.

[Veloso, 1992] Manuela Veloso. *Learning by Analogical Reasoning in General Problem Solving.* PhD thesis, CMU, 1992. CS Technical Report CMU-CS-92-174.